

第1次隨堂-隨堂-QZ1

學號：112111207

姓名：陳品霖

作業撰寫時間：30 (mins · 包含程式撰寫時間)

最後撰寫文件日期：2024/09/30

本份文件包含以下主題：(至少需下面兩項，若是有多者可以自行新增)

- [] 說明內容
- [] 個人認為完成作業須具備觀念

說明程式與內容

1. 請參閱投影片Topic0_1的第37頁，並於專案下開啟一個1.py的檔案，將該頁程式鍵入後，請再依照程式內的person1實例，再生成一個person2，並觀察和person1的差異

ans：

1. person1 和 person2 是 Person 類別的兩個不同實例，它們的 name 和 age 屬性各自獨立。
2. 雖然 person1 和 person2 都共享相同的類別方法 (greet 和 haveBirthday)，但這些方法在每個實例中操作的是該實例自身的屬性 (name 和 age)，因此它們的輸出會根據實例不同而變化。

```
# @package 物件導向設計
class Person:
    name: str
    age: int

    def __init__(self, name: str, age: int):
        self.name = name
        self.age = age

    def greet(self) -> str:
        return f"Hello, my name is {self.name} and I am {self.age} years old."

    def haveBirthday(self):
        self.age += 1
        print(f"Happy birthday, {self.name}! You are now {self.age} years old.")

# 創建 Person1 類的實例
person1 = Person(name="Alice", age=30)

# 呼叫 greet 方法
print(person1.greet()) # 輸出: Hello, my name is Alice and I am 30 years old.
```

```
# 呼叫 haveBirthday 方法
person1.haveBirthday() # 輸出: Happy birthday, Alice! You are now 31 years old.

# 呼叫 greet 方法
print(person1.greet()) # 輸出: Hello, my name is Alice and I am 31 years old.

# 創建 Person2 類的實例
person2 = Person(name="Ryan", age=23)

# 呼叫 greet 方法
print(person2.greet()) # 輸出: Hello, my name is Ryan and I am 23 years old.

# 呼叫 haveBirthday 方法
person2.haveBirthday() # 輸出: Happy birthday, Ryan! You are now 24 years old.

# 呼叫 greet 方法
print(person2.greet()) # 輸出: Hello, my name is Ryan and I am 24 years old.
```

個人認為完成作業須具備觀念

1. 物件導向設計，類別 (Class)，物件 (Object) 的概念，屬性 (Attribute)：類別中定義的數據（例如 name 和 age），每個實例都可以擁有不同的屬性值。
2. 方法 (Method)：類別中定義的行為，這些行為可以操作屬性。這裡的 greet 和 haveBirthday 方法就是類別方法。
3. 初始化方法 (**init**) 當實例被創建時，**init** 方法會自動被呼叫，並且可以接受參數來為屬性初始化值。在這裡，我們使用 **init** 方法來設置每個 Person 實例的 name 和 age。